Quality Considerations in Delay Fault Testing*

Alicja Pierzynska and Slawomir Pilarski

School of Computing Science, Simon Fraser University
Burnaby, BC, Canada V5A 1S6
e-mail: {alicja,pilarski}@cs.sfu.ca

Abstract

We examine delay models used in VLSI circuit testing. Our study includes electrical-level simulation experiments with HSPICE. We show phenomena which significantly affect the actual delays, but which are not taken into account by the existing models used in testing. Our analysis questions the test quality offered by test generation procedures used so far.

1 Introduction

Current test generation procedures that aim at delay faults in combinational circuits are based on the assumption that a fault can be detected using a pair of consecutive input vectors [1, 2, 3]. Such a pair of input patterns is referred to as a *test pair*, or simply a *test*. Test pairs are generated for various faults [4], which usually fall into one of the two categories: the *gate fault model* [5, 6], or the *path fault model* [7].

Given a path, test generation can be based solely on the logic structure of the circuit. Nonetheless, estimates of the actual delays play an important role in delay testing. For example, test generation for gate faults includes selection of the longest paths in the circuit [8]. In the case of path faults, ideally all path of the circuit are tested. In practice, however, the number of paths is often too large and test generation focuses only on the longest paths [9]. Estimates of gate delays are also used to evaluate the number of detected gate faults [6, 10]. In the path fault model, the number of detected faults is considered to be independent of the actual delays in the circuit. However, as noted in [1], this simplistic assumption may lead to a pessimistic evaluation.

Estimates of the actual delays necessary in test generation are based on various delay models. The simplest delay model assigns a single delay value to all circuit gates or to a particular gate type. More refined delay models take into account that the delay may depend on the fanout [9, 8], or on the transition propagated through the gate (rising or falling) [7, 5]. In [1], the authors present a delay model that includes gate inertia, i.e., the fact that the impulse at the input must be long enough to change the value at the output of a gate. They also note that, due to manufacturing variations, different physical instances of the same gate may have different delays. The generic version of their delay model can assign different delays to different input pairs. In practice however, the authors restrict their theoretical model and distinguish only between the rising and the falling delay.

In general, all currently used test generation procedures known to us assume the following:

> *The delay of a gate or the delay along a path depends on the type of the propagated transition (rising or falling), but is otherwise independent of the test applied to the circuit.*

The first indication that the assumption stated above does not hold in real CMOS circuits was given in [11] and independently documented in [12]: some deficiencies of delay models have been shown and the need for 3-vector tests has been demonstrated. This paper identifies further problems with delay models used in testing. It presents the dependency of gate delays on logic values and transitions at seemingly unrelated nodes of the circuit. Due to a more comprehensive analysis, some of our conclusions differ from those presented in [11].

The paper is organized as follows. Section 2 summarizes basic results presented in [11] and some conclusions which have not been explicitly pointed out in [11], but which are implied by the results presented therein. It also describes gate models and notation used in this paper. Sections 3 and 4 show unexpected delay effect of fan-outs: for stable logic values at fanout gates (Section 3) and for transitions at these gates (Section 4). Section 5 discusses how values and transitions at seemingly unrelated nodes of the circuit may change — increase or decrease — circuit delays. Finally, Section 6 concludes the paper.

2 Background

Fig. 1 presents transistor diagrams of a static two-input CMOS NAND gate and a static two-input CMOS NOR gate. These diagrams show circuit nodes which are not visible at the gate level, but which are important in timing considerations [11, 12]. These nodes are located between two serially connected transistors: in Fig. 1 they are labeled s. When both inputs of the gate take a *controlling value* [13] (0 for the NAND gate, 1 for the NOR gate), both of the

*This work was partially supported by grants from the Advanced Systems Institute of British Columbia and the Natural Sciences and Engineering Research Council of Canada.

serially connected transistors are cut off. Thus, the charge stored at node s is determined by the last input vector for which the gate inputs had at most one controlling value. This charge can significantly affect gate delays [11, 12]. Because of such a sequential behavior, we refer to this node as the *sequential node* of the gate.

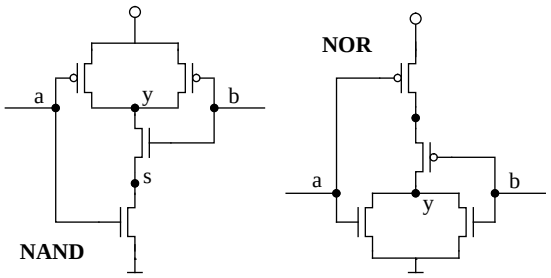

Figure 1: Two-input CMOS NAND and NOR gates.

Nodes visible at the gate-level, but considered unrelated to a path, can also affect the path delay. More specifically, due to various transistor capacitances [14, pages 125–129], some circuit nodes considered unrelated to the path are coupled with the path. Because of this, values or transitions at these nodes can significantly affect the actual path delay. This includes

1. non-path inputs of path gates.
2. non-path inputs of gates not included but connected to the path.

Although not stated directly, the influence of the nodes identified in (1) can be deducted from the results presented in [11]; a brief description is included in Section 2.3. The influence of values and transitions identified in (2) is studied in detail in this paper.

For brevity, we illustrate our observations with experimental results for two-input NAND gates, but some of our conclusions are based on experiments with both NAND and NOR gates [12].

2.1 Notation

From the electrical point of view, logic values at sequential nodes are not equivalent to logic values at gate outputs [14, pages 55-57]: in a NAND gate, a logic 0 at the sequential node corresponds to 0V ('good 0' [14]), while a logic 1 at this node corresponds to a voltage lower than 5V ('poor 1' [14]). In this paper, 'good 0' is abbreviated as *g0*, 'poor 1' is abbreviated as *p1*.

Since gate inputs are not identical from the electrical point of view [14], we distinguish between them in the following way: the inputs are labeled according to the order of serially connected transistors as shown in Fig. 1: a denotes the input connected to the transistor whose source is at the ground (NAND gate) or at the power (NOR gate), b denotes the other input. Letter y denotes the output. This node-labeling convention is used throughout the paper. Thus, for example, in a circuit which contains a gate referred to as *NAND189*, the nodes of this gate are referred to as *NAND189.a, NAND189.b, NAND189.s,* and *NAND189.y.*

To state that a node x of a gate takes logic value 0 (or 1), we write x=0 (or x=1). If a sequence of logic values $\lambda_1, \lambda_2, \ldots$ appears at node x, we write x=$\lambda_1\lambda_2\ldots$. We will also use $\uparrow$ to denote a rising transition ($0 \rightarrow 1$), and a $\downarrow$ to denote a falling transition ($1 \rightarrow 0$).

2.2 Gate Modeling

All of our simulation experiments were conducted using an analog circuit simulator HSPICE by Meta-Software, Inc. In these experiments, PMOS and NMOS transistors are assumed to be fabricated by 1.2μ CMOS process available to Canadian universities through Canadian Microelectronics Corporation (CMC). All transistors are of the same size: the length is 1.2μ (the minimal length) and the width is 3.2μ (determined by the size of the smallest contact). Delays are measured at the 50%-level of the power supply voltage, i.e., at the level of 2.5V.

2.3 Basic Gate Delays

	a	b	init. s	y	delay (ps)	ratio
1	$\downarrow$	$\downarrow$	g0	$\uparrow$	179.23	1
2	1	$\downarrow$	g0	$\uparrow$	289.42	1.615
3	$\downarrow$	1	g0	$\uparrow$	341.19	1.904
4	1	$\uparrow$	g0	$\downarrow$	223.53	1
5	$\uparrow$	1	p1	$\downarrow$	237.44	1.062
6	$\uparrow$	$\uparrow$	g0*	$\downarrow$	282.29	1.263
7	$\uparrow$	$\uparrow$	p1*	$\downarrow$	297.85	1.332

Table 1: NAND gate delays.

Tab. 1 presents delays observed in a NAND gate [12]. Column *init. s* shows the value at the sequential node immediately before the last input vector is applied. A star by the initial value at node s (as in 0* or 1*) indicates that this value was determined by a vector preceding the two-vector input sequence. Column *ratio* gives the ratio of a given delay to the minimum delay observed for the same transition at the gate output.

The experiment listed in row 1 detects a delay fault only if both input paths are affected [13]. Thus, it is usually not considered to be a test. From the delay point of view, this input pair is not desirable, since it results in a delay significantly shorter than any input pair listed in row 2 or 3.

A comparison of the remaining simulation results listed in Tab. 1 implies the following:

- For two identical two-vector tests, the delay may differ depending on vectors preceding the test [11, 12] (compare tests 6 and 7). Thus, with respect to delay, a logic gate behaves like a sequential circuit.

  The vector preceding the two-pattern test will be referred to as a *pre-initializing vector*. In general, pre-initialization is important for tests in which both input values undergo a simultaneous transition from a controlling to a non-controlling value.
- A gate delay is affected by logic values at non-path inputs of path gates (compare test 4 with any of tests 6 or 7). Tests with simultaneous input

transitions from a controlling to a non-controlling value result in a significantly higher delay than a test with a stable non-controlling value at the side fan-in node, *regardless of pre-initialization.*

The non-path inputs of path gates will be referred to as *side inputs of the path gates*, or simply *side fan-in nodes.*

3 Unexpected effect of fan-out

An important practical difference between single gates considered in [11] and many practical circuits lies in fan-out. The fact that delay depends on the number of fan-out gates is well-known [8, 9], but it has not been examined how logic values at the inputs of fan-out gates influence gate delays.

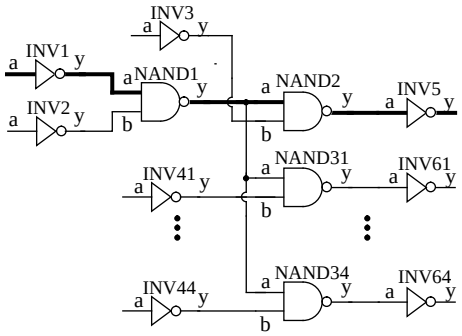

Figure 2: Gate NAND1 in *aa* connection.

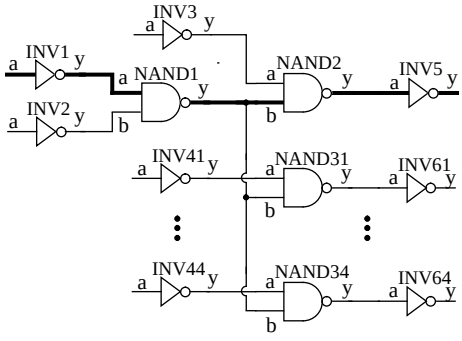

Figure 3: Gate NAND1 in *bb* connection.

Fig. 2 and Fig. 3 show circuits in which we conducted experiments with NAND gates. The inverters served as 'unit fan-in' and 'unit fan-out' gates. In these circuits, we consider a path indicated by a thicker line along gates NAND1 and NAND2. Gates NAND31–NAND34 do not belong to the path; they will be referred to as *side fan-out gates.* The non-path inputs of the side fan-out gates (e.g., NAND31.b, NAND34.b in Fig. 2) will be referred to as *side inputs of the side fan-out gates*, or simply *side fan-out nodes.*

The only difference between circuits presented in Fig. 2 and in Fig. 3 is that in Fig. 2 gate NAND1 is loading input *a* of the fan-out gates, whereas in Fig. 3, it is loading input *b* of those gates. For this reason, we will refer to the first circuit as being in *aa* connection, and to the second circuit as being in *bb* connection.

	NAND1 a	NAND1 b	NAND31..34 b	NAND31..34 init. s	delay (ps)	ratio
1	↑	1	0	g0	446.71	1.018
2	↑	1	1	g0	438.63	1
3	↓	1	0	g0*	665.95	1
4	↓	1	0	p1*	682.32	1.024
5	↓	1	1	p1	674.08	1.012

Table 2: Delays of gate NAND1 in *aa* connection.

	NAND1 a	NAND1 b	NAND31..34 a	NAND31..34 init. s	delay (ps)	ratio
1	↑	1	0	g0	372.57	1
2	↑	1	1	g0	439.95	1.181
3	↓	1	0	g0*	568.29	1
4	↓	1	0	p1*	568.29	1
5	↓	1	1	p1	652.77	1.148

Table 3: Delays of gate NAND1 in *bb* connection.

In the circuits described above, we observed the delay of the NAND1 gate for two values at side fan-out nodes: 0 or 1. The side input of NAND2 was assigned a non-controlling value. Tabs. 2 and 3 summarize the results.

For circuits in *aa* connection, the differences in the propagation time are relatively small: below 3% for a NAND gate (and below 6% for a NOR gate [12]). For the circuits in *bb* connection the differences are much bigger: up to 18% for a NAND gate (and up to 34% for a NOR gate [12]). The longest delays in *bb* connection occur when the side fan-out gates propagate the transition through, i.e., the value at side fan-out nodes is non-controlling. When the value at the side fan-out nodes is controlling, the delay is shorter. At the same time, there is hardly any dependency related to the value at the internal node of side fan-out gates.

Observation 1 *The actual delay of a gate may significantly depend on logic values at side fan-out nodes.*

In subsequent experiments we investigated gate delays as a function of the number of fan-out gates as well as logic values at side fan-in and side fan-out nodes.

The experiments were conducted for circuits in *bb* connection (Fig. 3), because these circuits manifest higher dependency of the delay on side fan-out values. We varied the number of side fan-out gates, and, as before, we recorded NAND1 delays for both logic values (0 and 1) at side fan-out nodes. In contrast to the experiments presented in Tabs. 2 and 3, we considered not only just one, but all possible input combinations corresponding to tests 2–7 discussed in Section 2.3 (Tab. 1).

The results are presented in Fig. 4 for the falling transition at the output of NAND1 gate, and in

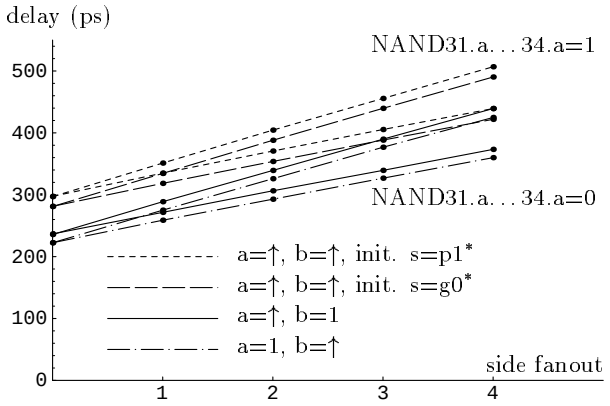

Figure 4: NAND1 in *bb* connection: delays for y=↓.

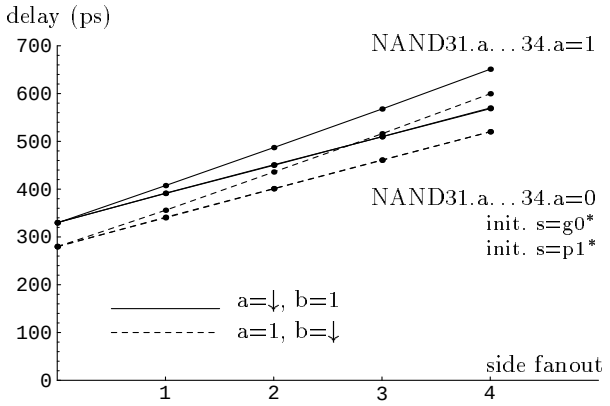

Figure 5: NAND1 in *bb* connection: delays for y=↑.

Fig. 5 for the rising transition at this output. Fig. 4 shows four pairs of plots: one pair for each combination of input values corresponding to tests 4–7 in Tab. 1. The upper line of every pair corresponds to the non-controlling value at side fan-out nodes, while the lower line corresponds to the controlling value at these nodes. Similarly, Fig. 5 shows plots corresponding to tests 2–3 in Tab. 1. In this case, for a controlling value at side fan-out nodes, we consider two possible values at the sequential node of side fan-out gates, but the results of these two cases are virtually identical.

Observation 2 *The delay is linear with respect to the number of fan-out gates (as assumed in some of the existing delay models [8]). However, in contrast to what has been commonly assumed to date, the coefficient of linearity is not a fixed value for a given input transition: it depends on the values at side fan-out nodes.*

4 Dynamic changes at fan-out gates

So far we have examined the influence of different stable values at side fan-out nodes. In real circuits, however, the values at side fan-out nodes may change when the test is being applied. To investigate how dynamic changes at side fan-out nodes influence the delay, we examined path delay for various transitions at side fan-out nodes in the circuit presented in Fig. 3. The experiments were conducted for path *NAND1.a–NAND2.y* while *NAND1.a*=↑, and then while *NAND1.a*=↓ (*NAND1.b*=1).

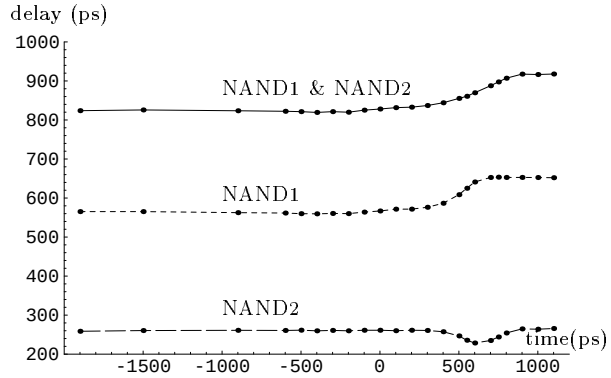

Figure 6: Gate delays vs. the time difference between *INV41.a...INV44.a*=↑ and *INV1.a*=↑.

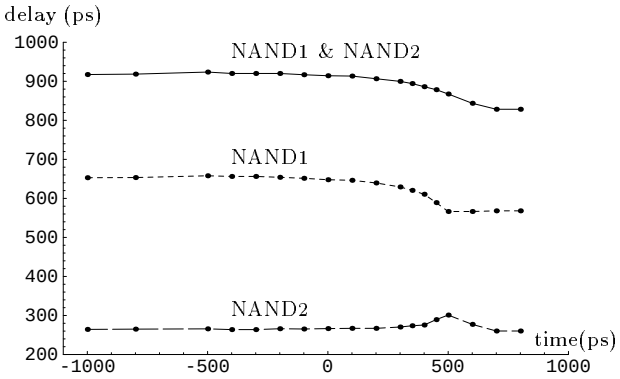

Figure 7: Gate delays vs. the time difference between *INV41.a...INV44.a*=↓ and *INV1.a*=↑ for *INV41.s...INV44.s*=g0.

The results of our simulation experiments are presented in Figs. 6–10. Each of these figures shows three plots: one for NAND1, one for NAND2, and one for the sum of delays associated with both gates.

In the case of NAND1.a=↓ (INV1.a=↑) the delay associated with NAND1 and NAND2 changes monotonously (see Figs. 6, 7, and 8). For NAND1.a=↑ (INV1.a=↓), however, the path delay can be longer or *shorter* than the delay for any stable value at side fan-out nodes (see Figs. 9 and 10); numerical results are presented in Tab. 4. As shown in this table, the maximum path delay for transitions at side fan-out nodes (*NAND31.a–NAND34.a*=↓) can be 9.8% longer than the maximum delay for stable values at these

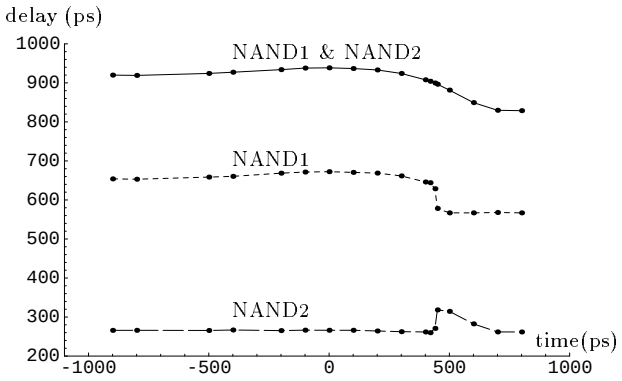

Figure 8: Gate delays vs. the time difference between *INV41.a...INV44.a*=↓ and *INV1.a*=↑ for *INV41.s...INV44.s*=p1.

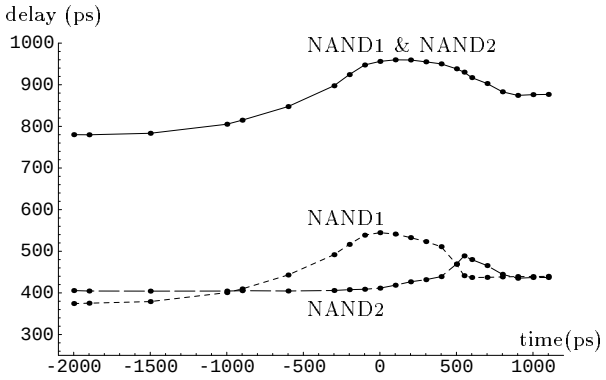

Figure 9: Gate delays vs. time difference between *INV41.a...INV44.a*=↑ and *INV1.a*=↓.

nodes (*NAND31.a–NAND34.a*=1). Furthermore, the minimum delay for transitions at side fan-out nodes (*NAND31.a–NAND34.a*=↑) can be 9% shorter than the minimum delay under stable values at these nodes (*NAND31.a–NAND34.a*=0). Note that the percentages refer to the path consisting of two gates (NAND1 and NAND2), whereas the delay change is mostly associated with one gate only (NAND1).

Observation 3 *Transitions at side fan-out nodes may result in considerably longer or shorter delays than under any stable values at side fan-out nodes.*

5 Opposite effect of transitions

As observed in in [11, 12] (see Section 2.3) simultaneous transitions from controlling to non-controlling values at fan-in nodes increase the path delay. Because of this, the authors of [11] suggest that efficient delay testing should involve high node activity. This surmise, however, is based on incomplete analysis of circuit phenomena. In fact, high node activity may

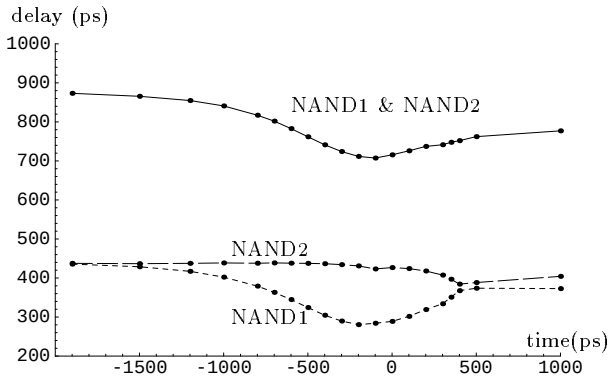

Figure 10: Gate delays vs. the time difference between *INV41.a...INV44.a*=↓ and *INV1.a*=↓.

	max (Fig. 9)	min (Fig. 10)
for stable values at side fan-out nodes	875.34	777.11
for transitions at side fan-out nodes	959.85	707.39
ratio	1.096	0.910

Table 4: The sum of NAND1 and NAND2 delays (ps): test comparison.

be undesirable if we consider the influence of values or transitions at side fan-out nodes.

To examine how high node activity may effect delays, we conducted several experiments with the circuit in Fig. 3. Let *NAND1.a*=↑. First consider a situation in which higher node activity results in longer delays, as suggested in [11]: for fixed values at the side fan-out nodes, the delay of path *NAND1.a–NAND2.b* is longer under test {*NAND1.a*=↑, *NAND1.b*=↑} than under test {*NAND1.a*=↑, *NAND1.b*=1}. Tests 1 and 2 in Tab. 5 illustrate it for a stable 1 at the side fan-out nodes. Now examine the same circuit with high node activity that involves not only transitions at side fan-in nodes, but also transitions at side fan-out nodes:

- A late ↑, or an early ↓, at side fan-out nodes may be equivalent to a stable 0 at these nodes, hereby decreasing the delay as compared to that for a

	NAND1 a	NAND1 b	NAND1 init. s	NAND 31..34.a	delay(ps) NAND1..2	ratio
1	↑	1	p1	1	876.25	1
2	↑	↑	p1*	1	947.39	1.081
3	↑	↑	p1*	0	846.67	0.966
4	↑	↑	p1*	↑	783.01	0.893

Table 5: Delays for varius values at side fan-in and side fan-out nodes.

stable 1 (see Section 3): the delay in test 3 is 10.6% shorter than in test 2.

- Transition ↑ at side fan-out nodes may decrease the delay even below that for a stable 0 at side fan-out nodes (see Section 4): the delay in test 4 (for which the time difference between *INV41.a*...*INV44.a*=↓ and *INV1.a*=↓ is -100ps) is 7.5% shorter than in test 3.

In both cases described above, the delay is not only shorter than the delay under test 2, it is also shorter than the delay under test 1, involving the *lowest* node activity. In particular, the delay under test 4 (highest node activity) is 10.7% shorter than the delay under test 1 (lowest node activity). Such a phenomenon may be especially important when logic values in a circuit are strongly dependent on each other. For example, if, in the circuit presented in Fig. 3, *NAND1.b*=↑ results in *NAND31.a*...*NAND34.a*=↑, then test 1 involving the *lowest* node activity turns out to be the most desirable.

Observation 4 *High node activity in a circuit under test may either significantly increase or significantly decrease path delay, depending on the type of transitions, their location and timing.*

6 Discussion and Conclusions

Tests which are commonly considered to be equivalent with respect to a particular fault may lead to different pass/fail decisions due to different gate delays. The cumulative differences in circuit delay for two supposedly equivalent tests can be very high. For example, in the case of a NAND gate with five fanout gates of the same type and a falling transition at the output, possible delays can vary as much as 41% for stable values at side fanout nodes (see Fig. 4 and compare delay under test {*a*=1, *b*=↑} in which *NAND31.a*... *NAND34.a*=0 with test {*a*=↑, *b*=↑, *initial s*=p1*} in which *NAND31.a*... *NAND34.a*=1). This difference can be even bigger for a ↑ transition at side fanout nodes (see Fig. 10).

To ensure the longest propagation times and to achieve the highest test quality, delay models used in test generation need to be extended. They should take into account that the actual circuit delays depend on the following factors:

- pre-initialization,
- inputs which propagate the transition,
- logic values and transitions at side fan-in and side fan-out nodes.

Contrary to the conclusion in [11], high node activity may be undesirable in delay testing. A simple, but not necessarily the most desirable, way to improve the quality of existing test procedures is a faster test application. More precisely, the time interval of the second pattern application should be shorter than that determined by the system clock by the ratio of the fastest to the slowest propagation time along a path.

A more sophisticated way to improve the test quality lies in new test generation procedures, which should take into account the phenomena discussed in this paper. Such test generation procedures need technology-dependent information, but the qualitative dependencies between circuit delays are most likely to be the same as in our experiments.

References

[1] J. L. Carter, V. S. Iyengar, and B. K. Rosen, "Efficient test coverage determination for delay faults," in *Proc. International Test Conference*, pp. 418–427, September 1987.

[2] J. Savir, "Skewed-load transition test: Part I, calculus," in *Proc. International Test Conference*, pp. 705–713, September 1992.

[3] S. Patil and J. Savir, "Skewed-load transition test: Part II, coverage," in *Proc. International Test Conference*, pp. 714–722, September 1992.

[4] A. Krasniewski and L. Wronski, "Coverage of delay faults: When 13% and 19% mean the same," in *Lecture Notes in Computing Science: Dependable Computing* (K. Echtle, D. Hammer, and D. Powell, eds.), vol. 852, pp. 178–195, Springer Verlag, 1994.

[5] A. K. Pramanick and S. M. Reddy, "On detection of delay faults," in *Proc. International Test Conference*, pp. 845–856, September 1988.

[6] V. S. Iyengar, B. K. Rosen, and J. A. Waicukauski, "On computing the sizes of detected delay faults," *IEEE Trans. CAD*, vol. 9, pp. 299–312, March 1990.

[7] G. L. Smith, "Model for delay faults based upon paths," in *Proc. International Test Conference*, pp. 342–349, November 1985.

[8] E. S. Park and M. R. Mercer, "An efficient delay test generation system for combinational logic circuits," *IEEE Trans. CAD*, vol. 11, pp. 926–938, July 1992.

[9] S. Patil and S. Reddy, "A test generation system for path delay faults," in *Proc. IEEE International Conference on Computer Design*, pp. 40–43, October 1989.

[10] Y. Aizenbud, M. Leibowitz, P. Chang, D. Smith, B. Koenemann, V. Iyengar, and B. Rosen, "AC test quality: Beyond transition fault coverage," in *Proc. International Test Conference*, pp. 568–577, September 1992.

[11] P. Franco and E. J. McCluskey, "Three-pattern tests for delay faults," in *IEEE VLSI Test Symposium*, pp. 452–456, April 1994.

[12] A. Pierzynska and S. Pilarski, "Timing issues in delay fault testing," Tech. Rep. CSS/LCCR TR94-03, Simon Fraser University, April 1994.

[13] K. T. Cheng and H. C. Chen, "Delay testing for non-robust untestable circuits," in *Proc. International Test Conference*, pp. 954–961, October 1993.

[14] N. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*. Massachusetts: Addison-Wesley, 1985.

[15] M. Favalli, M. Dalpasso, P. Olivo, and B. Ricco, "Analysis of dynamic effects of resistive bridging faults in CMOS and BiCMOS digital ICs," in *Proc. International Test Conference*, pp. 865–874, October 1993.